

CHAPTER – 2

Concepts of Object Oriented Programming.(Expected Marks = 4)

(ഈ നോട്ട്സിൽ വിവിയോയും ഉൾപ്പെടുത്തിയിട്ടുണ്ട്. പാഠഭാഗങ്ങൾക്കൊപ്പം വിവിയോ ട്യൂട്ടോറിയലുകളും കാണാവുന്നതാണ്. മൊബൈൽ ഫോണിലാണ് നിങ്ങൾ ഈ നോട്ട്സ് വായിക്കുന്നതെങ്കിൽ QR കോഡിൽ ക്ലിക്ക് ചെയ്യുക. Printed Note ആണെങ്കിൽ QR കോഡ് സ്കാൻ ചെയ്യുക.)



**Programming Paradigm**  
(പ്രോഗ്രാമിംഗ് മാതൃക)

**1. Procedure Oriented Programming. (POP)**

( Click or Scan here to see the video)

It gives more importance to function than data.

(ഡാറ്റയെക്കാൾ പ്രാധാന്യം ഫങ്ഷൻ കൊടുത്തുകൊണ്ടുള്ള പ്രോഗ്രാമിംഗ് രീതിയാണിത്.)



**2. Object Oriented Programming. (OOP)**

It gives more importance to data.

(ഡാറ്റയ്ക്ക് കൂടുതൽ പ്രാധാന്യം കൊടുത്തുകൊണ്ടുള്ള പ്രോഗ്രാമിംഗ് രീതിയാണിത്.)

**POP v/s OOP**

POP	OOP
Data is given less importance. (ഡാറ്റയ്ക്ക് പ്രാധാന്യം കുറവാണ്.)	Data is given more importance. (ഡാറ്റയ്ക്ക് കൂടുതൽ പ്രാധാന്യം നൽകുന്നു.)
Function is given more importance. (ഫങ്ഷൻ കൂടുതൽ പ്രാധാന്യം നൽകുന്നു.)	Function is given less importance. (ഫങ്ഷൻ പ്രാധാന്യം കുറവാണ്)
Creating new data types is difficult. (പുതിയ ഡാറ്റ ടൈപ്പുകൾ സൃഷ്ടിക്കുന്നത് ബുദ്ധിമുട്ടാണ്.)	New data types can easily defined. (പുതിയ ഡാറ്റ ടൈപ്പുകൾ എളുപ്പത്തിൽ നിർവചിക്കാൻ കഴിയും.)
Poor real world modelling. (യഥാർത്ഥ സാഹചര്യമായി ബന്ധം കുറവാണ്.)	Easy to define real world situation. (യഥാർത്ഥ സാഹചര്യം നിർവചിക്കുവാൻ എളുപ്പമാണ്.)

**Basic Concepts of OOP**

( Click or Scan on the QR Code to see the video)



**(1) Class**

- A Class is a blueprint of objects that defines data and functions .  
(ഡാറ്റയും ഫങ്ഷനും അടങ്ങിയ ഒബ്ജക്ടിന്റെ വിശദരൂപമാണ് Class )
- It represents the set of properties or methods that are common to all objects.  
(എല്ലാ object കൾക്കും പൊതുവായുള്ള properties ഉം methods ഉം നിർവചിച്ചിരിക്കുന്നത് class ൽ ആണ്.)

**(2) Object**

- Object is an instance of class. Objects are created from class.  
(object കൾ നിർമ്മിക്കുന്നത് class ൽ നിന്നാണ്. class ൽ ഉള്ള എല്ലാ ഡാറ്റയും ഫങ്ഷനുകളും object ൽ അടങ്ങിയിരിക്കുന്നു.)

**Class & Objects**

Example :

```

Class student
{
    Private :
        int rollno;
        char name(20);
    Public :
        void read( );
        void print( );
};
    
```

} Data

} Member Functions



```
student s1, s2, s3 ;
```



student എന്ന class ൽ നിന്നും s1, s2, s3 എന്ന മൂന്നു object കൾ ഉണ്ടാക്കിയിരിക്കുന്നു.

( Click or Scan on the QR Code to see the video)



### (3) Data Abstraction

- It refers to the act of showing essential features hiding the background details or explanation.

(പശ്ചാത്തല വിശദാംശങ്ങളോ വിശദീകരണങ്ങളോ മറച്ചുവെച്ചു കൊണ്ട് അവശ്യ വിവരങ്ങൾ മാത്രം പ്രദർശിപ്പിക്കുന്നതിനെ Data Abstraction എന്ന് വിളിക്കുന്നു)

### (4) Data Encapsulation

- It refers to the binding of data and functions together and keeps them safe.

(ഇത് ഡാറ്റയെയും ഫംഗ്ഷനുകളെയും പരസ്പരം ബന്ധിപ്പിക്കുകയും അവയെ സുരക്ഷിതമായി സൂക്ഷിക്കുകയും ചെയ്യുന്നതിനെ സൂചിപ്പിക്കുന്നു.)

### (5) Polymorphism

- It is the ability to exist in many forms.

(പല രീതികളിൽ നിലനിൽക്കാനുള്ള കഴിവിനെ സൂചിപ്പിക്കുന്നു.)

### (6) Inheritance

- It is the ability of a class to inherit the properties of another class is known as Inheritance.

(ഒരു class ന്റെ സവിശേഷതകൾ മറ്റൊരു class നു കൈമാറാനുള്ള കഴിവിനെ inheritance എന്ന് വിളിക്കുന്നു.)

**Types of Inheritance** (വിവിധ തരത്തിലുള്ള inheritance കൾ)

1. Single Inheritance
2. Multiple Inheritance
3. Multilevel Inheritance
4. Hierarchical Inheritance
5. Hybrid Inheritance



www.rvgirls.com

### (7) Modularity

- It is the process by which a larger program is sub-divided into smaller programs called modules.

(ഒരു വലിയ പ്രോഗ്രാമിനെ മൊഡ്യൂളുകൾ എന്ന് വിളിക്കുന്ന ചെറിയ പ്രോഗ്രാമുകളായി വിഭജിക്കുന്ന പ്രക്രിയയാണിത്.)